

Short Papers

Microfluidic Injector Models Based on Artificial Neural Networks

Ryan Magargle, James F. Hoburg, and Tamal Mukherjee

Abstract—Lab-on-a-chip (LoC) systems can be functionally decomposed into their basic operating devices. Common devices are mixers, reactors, injectors, and separators. In this paper, the injector device is modeled using artificial neural networks (NNs) trained with finite element simulations of the underlying mass transport partial differential equations (PDEs). This technique is used to map the injector behavior into a set of analytical performance functions parameterized by the system's physical variables. The injector examples shown are the cross, the double-tee, and the gated-cross. The results are four orders of magnitude faster than numerical simulation and accurate with mean square errors (MSEs) on the order of 10^{-4} . The resulting NN training data compare favorably with experimental data from a gated-cross injector found in the literature.

Index Terms—Electrokinetic, injector, lab-on-a-chip (LoC), microfluidic, neural network, simulation.

I. INTRODUCTION

Microfluidic lab-on-a-chip (LoC) systems have been studied for more than a decade and have many applications in biology, medicine, and chemistry [1], [2]. They generally perform chemical analysis involving sample preparation, mixing, reaction, injection, separation analysis, and detection. Compared to traditional analytical labs, LoC has the significant advantage of increased analysis speed, parallel processing, and high integration and automation.

The simulation of complex LoC systems requires computationally expensive numerical solutions to partial differential equations (PDEs). As the design of LoC systems requires many repeated simulations, iterative design using numerical simulation is computationally infeasible. A much more efficient alternative involves functional decomposition into a series of interconnected blocks, as previously proposed for the mixer, injector, and separator [3]–[5], which can be used to compose an entire LoC [6]. Behavioral artificial neural network (hereafter referred to as simply neural network, or NN) modeling makes such an efficient decomposition possible. For the mixer and separator, band shape assumptions are used with analytical techniques to simplify the PDEs into several ordinary differential equations. For the injector, the desired output quantity is a scalar, and not a vector, as in the mixer and separator; hence, we use numerical NN techniques instead.

A diagram representing a typical LoC consisting of a mixer, reactor, injector, and separation system is shown in Fig. 1. There are two stages to its operation. In the first stage, a voltage is applied to nodes 1, 2, 3, 4, and 5. The resulting electric fields induce electrokinetic flow [7], which causes the fluid to be pumped through the channels. This

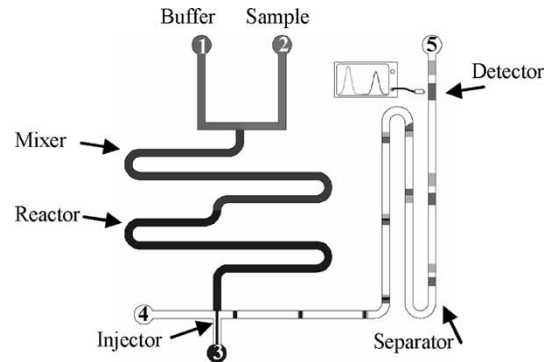


Fig. 1. Diagram of a canonical LoC.

dilutes the sample from node 2 with buffer from node 1 in the mixer before it is reacted with a catalyst and ultimately focused by nodes 4 and 5. In the second stage, called electrophoretic separation, new voltages are applied to nodes 1–5, which causes a band of the reacted analyte to be injected into the separation system. Because the analyte can be composed of multiple species of different charge, the individual species migrate at different speeds in the electric field causing them to separate from each other.

The importance of the injector as a component in a microfluidic separation system derives from the fact that it defines the shape and quantity of analyte that will be used for separation and analysis. Various forms of microfluidic electrokinetic injectors, such as the tee, the double-tee, the cross, the double-cross, and the gated-cross, have been introduced [8]–[12]. A first generation injector model produced by the authors was specific to the cross injector and was defined by a two-dimensional (2-D) parameter space [4]. This work improves on previous generation injector modeling by using NN behavioral modeling concepts to create a library of models for injectors, including the cross, the double-tee, and the gated-cross, each defined by a four-dimensional (4-D) parameter space. The methodology is not specific to any one injector topology and has been used in macromodels that have many more dimensions [13].

The methodology described here involves an exploration of a relevant portion of the injector physical parameter space using numerical solutions of the convection diffusion equation. These solutions are then used to train a NN that describes the performance of the injector. The behavioral output parameters are sufficient to construct a Gaussian distribution, approximating the width-averaged plug concentrations as a function of longitudinal position, for the input to the separation channel. Once the NN has defined the behavioral model, an explicit analytic equation defining the NN is created and can easily be ported into any modeling environment. This modeling method is related to the work done by [14], however, in our method, we have not utilized quasi-random training sequences, because our training data sets are not large enough to achieve the asymptotic low-discrepancy distribution [15]. Furthermore, our system has a larger number of physical variables making the computational cost of our network training greater, thus, we use the Buckingham- π theorem to reduce the size of the variable space needed for training simulations [16].

The scope of this work is primarily focused on the application of composable system simulation for continuous flow microfluidic LoC systems. The broad range of macromodeling application of NNs from

Manuscript received February 15, 2005; revised June 15, 2005. This work was supported by the Defense Advanced Research Projects Agency under the Air Force Research Laboratory, Air Force Material Command, USAF, under Grant F30602-01-2-0587, and the NSF ITR program under Grant CCR-0325344. Computing resources were made possible by the NSF Equipment Grant CTS-0094407 and Intel Corporation. This paper was recommended by Associate Editor K. Chakrabarty.

The authors are with the Carnegie Mellon University, Pittsburgh, PA 15213 USA.

Digital Object Identifier 10.1109/TCAD.2005.855936

Report Documentation Page				Form Approved OMB No. 0704-0188	
Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.					
1. REPORT DATE 15 JUN 2005		2. REPORT TYPE		3. DATES COVERED 00-00-2005 to 00-00-2005	
4. TITLE AND SUBTITLE Microfluidic Injector Models Based on Artificial Neural Networks				5a. CONTRACT NUMBER	
				5b. GRANT NUMBER	
				5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S)				5d. PROJECT NUMBER	
				5e. TASK NUMBER	
				5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Carnegie Mellon University, Department of Electrical and Computer Engineering, Pittsburgh, PA, 15213-3890				8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)				10. SPONSOR/MONITOR'S ACRONYM(S)	
				11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution unlimited					
13. SUPPLEMENTARY NOTES					
14. ABSTRACT see report					
15. SUBJECT TERMS					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT Same as Report (SAR)	18. NUMBER OF PAGES 8	19a. NAME OF RESPONSIBLE PERSON
a. REPORT unclassified	b. ABSTRACT unclassified	c. THIS PAGE unclassified			

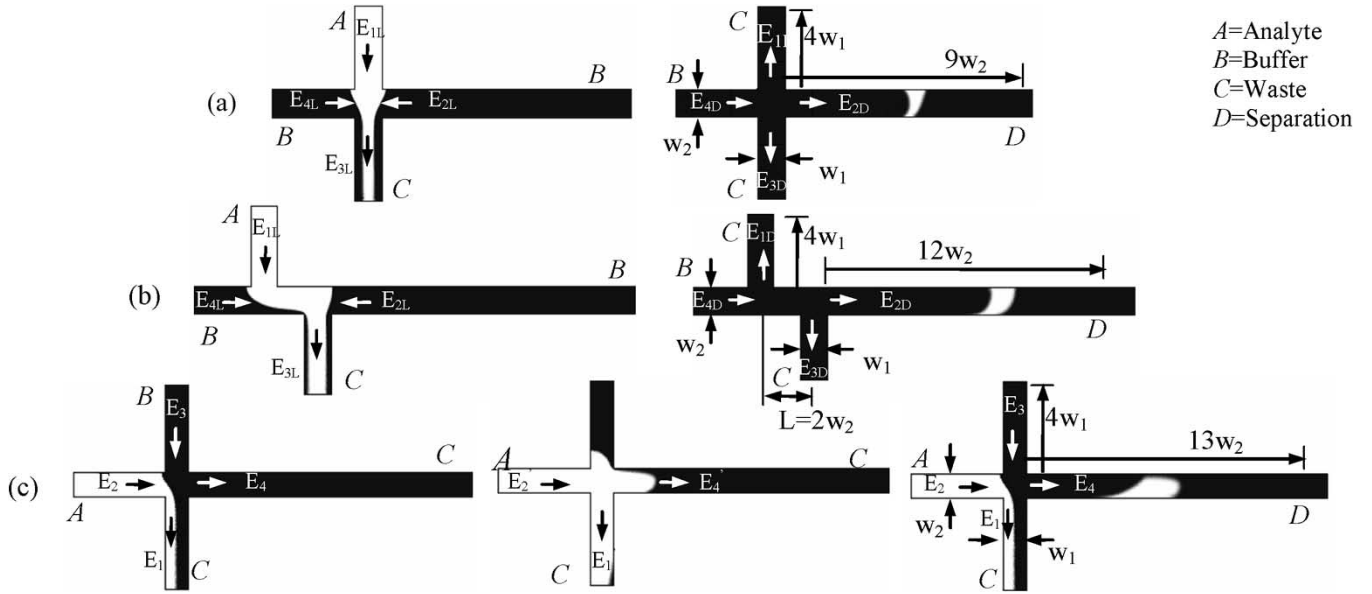


Fig. 2. Example injections and geometries for (a) the cross, (b) the double-tee, and (c) the gated-cross injectors. The light shades indicate regions of high analyte concentration.

very large scale integrated (VLSI) computer-aided design (CAD) [13], microfluidics [14], and radio frequency (RF) microelectromechanical system (MEMS) [17] suggests that further application in microfluidics, such as digital microfluidics, should be possible, but the exploration of such possibilities are beyond the scope of this work.

The injector topologies will be introduced in the next section. Section III will introduce the method used to model the injector device. The physical variables of the injectors and their nondimensional reduction will be shown in Sections III-A and III-B, and the parameters governing the creation of the NN model will be shown in Sections III-C and III-D. Finally, the impact of the resulting model on a composable system simulation will be shown in Section III-E, with conclusions in Section IV.

II. INJECTOR TOPOLOGIES

The three injectors considered in this paper are as follows: 1) the cross; 2) the double-tee; and 3) the gated-cross. Each injector is defined by its geometry and operating procedure. The geometries and operating steps are shown in Fig. 2.

The cross injector [Fig. 2(a)] is a simple intersection of two channels of the same width. The injector is operated by first loading the sample into the injection chamber, followed by a change in voltage that dispenses a portion of the injected flow into the separation channel. In both the loading and dispensing stages, accessory fields can be used to manipulate the shape of the injected plug. In the loading stage, accessory pinching (from E_{2L} and E_{4L}) can be used to achieve a more compact plug and to reduce the electrokinetic bias [8]. In the dispensing stage, accessory pullback (from E_{1D} and E_{3D}) prevents leakage into the separation channel.

The double-tee injector [Fig. 2(b)] is a more general form of the cross injector where the loading channels are offset to produce a larger injection plug. The operation scheme is the same as with the cross, with two stages of operation and optional accessory fields for plug shaping.

The gated-cross injectors [Fig. 2(c)] has the same geometry as the cross but uses a three-stage operating scheme. The gated-cross control scheme allows for continuous flow injection so that new sample can be loaded at the same time that previously dispensed plugs are run through the separation channel. The first step of operation involves

creating the gate in the injection chamber by counterflowing an analyte stream against a buffer stream. The second step involves removing the applied potential from the buffer port, which allows a portion of analyte to overflow the injection chamber. The final step is a return to the applied potentials of the first stage, which reestablishes the gate, while simultaneously injecting the overflowed analyte into the separation channel. Since the action taken in the second stage involves floating only one node while all others remain unchanged, no independent parameters are introduced by this stage. The fields and flow patterns are completely determined by the parameters set in the first stage.

III. MODELING METHODOLOGY

The complexity of the geometry and the resulting transitional electric field structure [18] makes behavioral models of the detailed physical performance of the injectors very difficult. The modeling approach used in this paper avoids the complexity of the underlying physical structure while maintaining an accurate description of the injector performance based upon the key characteristics that define the input/output mapping.

The injector modeling methodology consists of four steps. 1) The Buckingham- π theorem is used to reduce the order of the space of physical parameters describing the behavior of the injector to a reduced set of nondimensional parameters. In all three of the injectors shown in this paper, the number of nondimensional parameters describing the injector's operation is reduced to four, which is relatively small compared to the dimensions of typical VLSI analog design spaces [13]. 2) Numerical simulations are carried out at points in the nondimensional parameter space. The goal is to use as few numerical simulations as possible, since they are computationally expensive. 3) An NN is constructed to analytically describe the parameter space. Functions available in MATLAB 7 [19] were used to train the NN. A feed-forward back-propagation NN topology was chosen for its strong nonlinear regression capabilities and its ability to be converted to an explicit algebraic function. The NN as a regressor has the ability to handle high-dimensional spaces with relatively sparse data sets [20]. The NN also "learns" the functional mapping without any knowledge of the underlying physics or basis functions. 4) The

TABLE I
SUMMARY OF NONDIMENSIONAL PARAMETERS: THE TOP FOUR
COLUMNS ARE DYNAMIC AND THE BOTTOM TWO ARE GEOMETRIC

Cross	Double-Tee	Gated-Cross
$\pi_{1C}=E_{2,4L}/E_{1L}$	$\pi_{1D}=E_{2,4L}/E_{1L}$	$\pi_{1G}=E_2/E_1$
$\pi_{2C}=E_{1,3D}/E_{2D}$	$\pi_{2D}=E_{1,3D}/E_{2D}$	$\pi_{2G}=E_3/E_1$
$\pi_{3C}=\mu E_{3L}w_2/\kappa$	$\pi_{3D}=\mu E_{3L}w_2/\kappa$	$\pi_{3G}=\mu E_3w_2/\kappa$
$\pi_{4C}=\mu E_{2D}w_2/\kappa$	$\pi_{4D}=\mu E_{2D}w_2/\kappa$	$\pi_{4G}=w_2/\mu E_3T_{LD}$
$\pi_{5C}=w_1/w_2$	$\pi_{5D}=w_1/w_2$	$\pi_{5G}=w_1/w_2$
	$\pi_{6D}=L/w_2$	

trained network is converted to an explicit algebraic function, suitable for use in any software environment for simulation or synthesis. This conversion is done with a straightforward parsing algorithm, which can be applied to any generalized feed-forward topology. Thus, the main contribution of this paper is a method for creating microfluidic device models using a reduced set of nondimensional variables and input/output interface parameters.

A. Defining the π -Space

Implementation of the Buckingham- π theorem to the cross model is described in [4]. Extension of the results to the three injectors described here is summarized in Table I.

For all three injector topologies, the dynamic parameters, $\pi_{1(C,D,G)}$ through $\pi_{4(C,D,G)}$, are the independent variables used to create the models. The geometric parameters are held constant, such that $\pi_{5(C,D,G)} = 1$ and $\pi_{6D} = 2$. These geometries are representative of the vast majority of designs fabricated to date for each type.

Physically, the cross and the double-tee are very similar. From a synthesis point of view, the cross is simply a special case of the double-tee, where $\pi_{6D} = 0$. For both of these topologies, the $\pi_{1(C,D)}$ and $\pi_{2(C,D)}$ parameters describe the ratio of the accessory fields to the driving fields. For the loading stage, this describes the amount of pinching that is applied to the incoming analyte stream, and for the dispensing stage, this describes the amount of pullback applied to the dispensed band [4]. The $\pi_{3(C,D)}$ and $\pi_{4(C,D)}$ parameters describe the Peclet number for each stage.

The gated-cross has a set of parameters that differ from those of the cross and the double-tee. The first parameter π_{1G} represents the extent to which the gate is closed. As discussed in [12], as long as $E_1 \geq E_2$, the gate will remain closed in the limit of no diffusion. As π_{1G} is reduced, the gate is further closed, so in practice, $E_1 > E_2$. The second parameter represents the ratio of the buffer electric fields (E_3, E_4) to the analyte electric fields (E_2, E_1). As π_{2G} increases, the buffer fields become larger relative to the analyte fields. The third parameter π_{3G} represents the Peclet number of the loading phase. The final dynamic parameter π_{4G} is a measure of the ratio of length of the injected plug, as related to the floating time T_{LD} , to the channel width.

A significant concern in the operation of a gated-cross injector is the leakage that can occur in the separation channel if the gate is not sufficiently closed, as seen in Fig. 3. If the leakage is too great, the injector will not operate, because the increased noise floor will make a separation impossible. A region of feasibility must be determined within which the injector can be modeled. This leakage was analyzed in [12] as a function of the gate closure and the system Peclet number. They determined a boundary indicated by 1% leakage of the flux of the analyte from the source reservoir into the separation channel. In this work, we determine the boundary for a more complete set of physical parameters to create a region of feasibility.

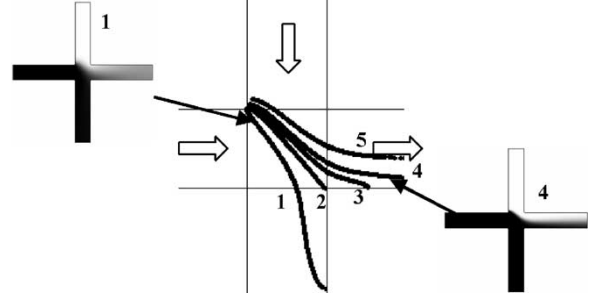


Fig. 3. Gated-cross leakage tested at Peclet numbers from 19 at contour 1 to ∞ at contour 5. The contours are defined by the 7% of the maximum concentration and agree very well with the numerical and experimental results found in [12].

B. Simulating the π -Space

The injectors were simulated in FEMLAB using the convection diffusion equation [21] given as

$$\frac{\partial c}{\partial t} + \vec{u} \cdot \vec{\nabla} c = \kappa \nabla^2 c \quad (1)$$

where c is the concentration distribution, \vec{u} is the velocity field, and κ is the diffusion coefficient.

To solve for the electrokinetic flow, first, Laplace's equation is solved for the electric potential as

$$\nabla^2 \Phi = 0 \quad (2)$$

which is used to determine the velocity, which is proportional to the electric field, through

$$\vec{u} = -\mu \vec{\nabla} \Phi \quad (3)$$

where μ is the electrokinetic mobility, and the negative gradient of the electric potential, Φ , is the electric field.

In all simulations, the channels are long enough to contain the transitions in the electric field structure. The buffer, analyte, and waste channels were all of the same length, $4w$, for each injector topology. The separation channel length, set so that it can encompass the entire injected band, is $9w$ for the cross, $12w$ for the double-tee, and $13w$ for the gated-cross. This will allow the numerical simulation data to be used to train the NN to contain a highly accurate description of the transitional electric field structure and its effect on the dispersion of the analyte.

For each injector topology, using conservation of current and assuming that all channel widths are the same, a relationship between the steady electric fields can be determined. As in prior work on the cross injector [4], the electric field magnitudes in the cross and the double-tee are related for the loading stage by

$$E_{3L} = E_{1L} + 2E_{2,4L} \quad (4)$$

and for the dispensing stage by

$$E_{4D} = E_{2D} + 2E_{1,3D} \quad (5)$$

where it is assumed that the accessory fields are applied symmetrically, i.e., $E_{2L} = E_{4L} = E_{2,4L}$ and $E_{1D} = E_{3D} = E_{1,3D}$. The fields for the gated-cross are related by

$$E_2 + E_3 = E_4 + E_1. \quad (6)$$

TABLE II
 π -SPACE DOMAINS FOR THE TRAINING SIMULATIONS,
 COVERING A WIDE RANGE OF PHYSICALLY VIABLE VALUES

	π_1	π_2	π_3	π_4
Cross	[0:1/2]	[0:2]	[50:5000]	[50:5000]
Double-Tee	[0:1/4]	[0:4]	[50:5000]	[50:5000]
Gated-Cross	[1/10:1]	[1:4]	[50:5000]	[1/20:1/3]

Using (4)–(6), and the conversion matrix introduced in [4], it is possible to calculate the electric potentials for the boundary conditions, as they change throughout the π -space.

To perform, on average, 200–300 simulations per injector, a FEMLAB script was written to automate the setup and execution of the simulations. The π -space domains for each of the injectors are defined in Table II. A parallel computing cluster was utilized, which reduced several weeks of computation to several days on a shared 100-node Beowulf cluster. This is a one-time computational expense used to create the much more efficient NN description, which “learns” from the results of the simulations.

The FEMLAB algorithms have been validated against many experiments found in the literature demonstrating microfluidic mixing, joule heating, injection, and separation [3], [5], [22]. To validate the simulations for the new gated-cross topology, a test was run to measure the leakage of the injector during the loading stage, as was first done in [12]. Fig. 3 shows the results of the simulations at the same parameters found in [12]. The resulting isoconcentration contours agree very well with [12] numerical and experimental results.

After numerically simulating points throughout the parameter space, the form of the output of the behavioral model is chosen. Analysis of the numerical results can be done to extract any desired behavioral outputs. In this work, the outputs were chosen to be the peak height and variance of an effective Gaussian describing the transversely averaged concentration profile output from the injector, such that the area and variance of the original distribution are conserved. The resulting form with nondimensional outputs is

$$\bar{f}(\pi_{1(k)}, \pi_{2(k)}, \pi_{3(k)}, \pi_{4(k)}) = \left[\frac{\sigma^2}{w^2} \frac{C_{\text{eff}}}{C_o} \right]_{(k)} \quad (7)$$

where $k \in [C, D, G]$, and C_o is the value of the uniform concentration of the input species from the reactor.

While the NN is inherently capable of describing multiple-input multiple-output models, we use a multiple-input single-output NN to train each output separately.

Using effective Gaussian outputs as the interface parameters for a system simulation accomplishes two purposes, namely: 1) the system simulation is efficient; and 2) while there is sufficient information to accurately describe the band after it has traveled a short distance down the separation channel, there is not enough information to describe the non-Gaussian shape of the band with high accuracy immediately after it leaves the injector.

As long as diffusion transversely homogenizes the band before detection occurs, or before another source of dispersion, such as a channel bend, is introduced, the approximation will provide accurate results. By requiring the time it takes particles to diffuse across the channel to be shorter than the time it takes the particles to reach the first channel bend or detector, a bound is defined as

$$\tau = \frac{Pe_w}{L} < 2 \quad (8)$$

where Pe_w is the Peclet number based on the channel width w and L is the length of straight channel before the detector or first channel

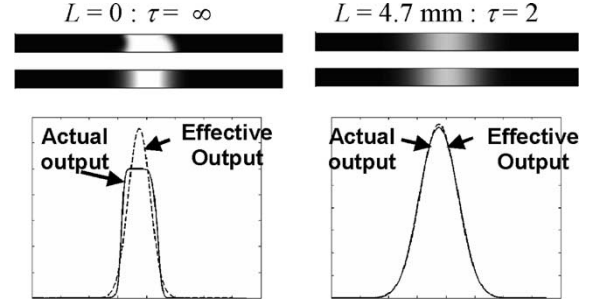


Fig. 4. Actual output of the double-tee injector (top channel) compared to effective parameterized output of the analytical model (bottom channel). The band on the right traveled 4.7 mm, which is when $\tau = 2$. For these simulations, $\pi_{1D} = 186$, $\pi_{2D} = 186$, $\pi_{3D} = 1/8$, and $\pi_{4D} = 0.57$.

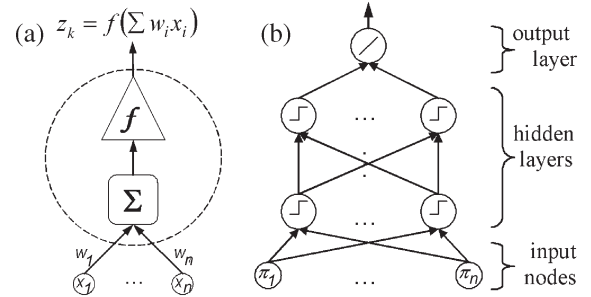


Fig. 5. (a) Activation z_k of a node is a function of the weighted inputs of the previous layer's activations. (b) General feed-forward NN topology utilizing multiple hidden layers with an unspecified number of nodes with nonlinear activation functions and a single output node using a linear activation function.

bend. This expression is conservative in that it requires the particles travel across the entire channel width to homogenize the band. In many cases, as seen in Fig. 2, the band coming out of the injector is axially compact and fairly uniform across the channel and so homogenization occurs much more rapidly.

An example for a double-tee injection is shown in Fig. 4. The picture on the left shows the band immediately after the injector, and the picture on the right shows the band after traveling 4.7 mm where, from (8), $\tau = 2$. As an example of an actual experimental system, the LoC developed by Chiem and Harrison [23] has $\tau = 0.2$, which is an order of magnitude within the bounds defined by (8).

C. Training the NN

An NN is a mathematical structure that is adept at learning nonlinear functional relations and complex item categorizations [20]. The general principle of operation of the feed-forward NN is shown in Fig. 5(a), and the topology of the two-layer NN used to model the injectors for this work is shown in Fig. 5(b). The feed-forward NN is a set of nodes that are connected only to the layers above and below. Each node takes, as the argument of an activation function, the weighted sum of the outputs of the layer below. To create an NN model, the number of layers, nodes per layer, node transfer function, and interconnecting weights must be selected.

It has been shown that to perform nonlinear regression to an arbitrary degree of accuracy, where the number of nodes is not an issue, two layers are sufficient (one hidden, one output) [24]. Likewise, with classification networks, it has been shown that a decision boundary of arbitrary complexity can be defined using only three layers (two hidden, one output) [20]. Therefore, our regression models use two-layer topologies, and our classification networks use three-layer topologies.

TABLE III
NEURAL NETWORK MSE AND CROSS VALIDATION: FOR ALL THREE INJECTORS, BOTH VALIDATION PARAMETERS ARE VERY SMALL

	MSE		KFCV		
	σ^2/w^2	C_{\max}/C_0	k	σ^2/w^2	C_{\max}/C_0
Cross	1.2e-4	2e-5	10	2e-6	9e-5
Double-Tee	1.8e-5	1e-5	8	3e-6	2e-5
Gated-Cross	4e-5	4.2e-5	10	1.1e-4	2.4e-4

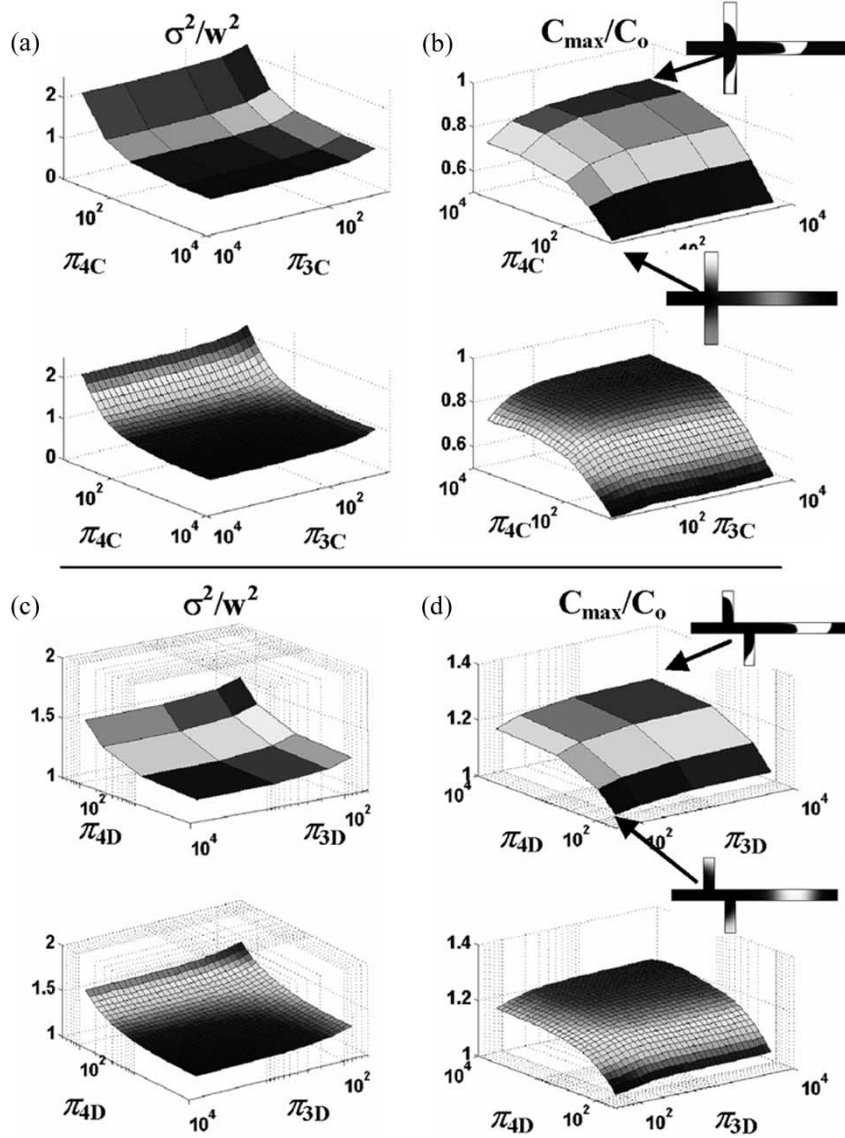


Fig. 6. (a) Variance and (b) effective peak concentration results for the cross, where π_{1C} and π_{2C} are fixed at 0.2. (c) Variance and (d) effective peak concentration results for the double-tee, where π_{1D} and π_{2D} are fixed at 0.05 and 0.3, respectively. The top plots show the FEMLAB simulations at a coarse set of points and the bottom plots show the much faster and higher resolution evaluations produced by the NN.

If the network is to perform nonlinear regression, the hidden layer consists of bound nonlinear activation functions, and the output nodes are unbound linear activation functions. If the network makes discrete classifications, then all nodes are bound nonlinear functions. In both cases, the actual shape of the nonlinear function is not important as long as it is bound [20]. The function used in this work is written as

$$f(x) = \frac{2}{1 + \exp(-2x)} - 1. \quad (9)$$

The number of input nodes is determined by the number of independent parameters in the model and the number of nodes in the output layer is determined by the number of dependent parameters in the model. The number of nodes in the hidden layer is arbitrary. Nodes are added or subtracted until the performance reaches a threshold. The hidden layers of the injector NNs contain 25 nodes.

After the topology and initial node count have been determined, the network is trained on the FEMLAB simulation data. The process of training an NN involves modifying the weights between layers until the outputs of the network sufficiently match the simulation

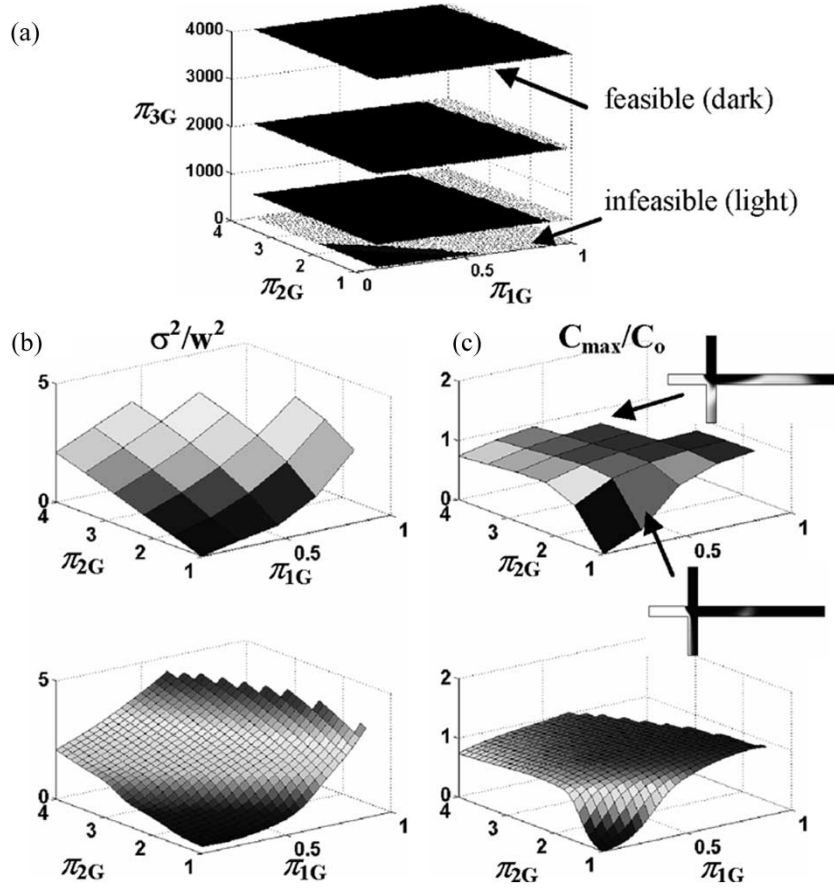


Fig. 7. (a) Gated-cross feasibility space (dark is feasible). (b) Variance and (c) effective peak concentration results for the gated-cross, where π_{3G} and π_{4G} are fixed at 232 and 0.1208, respectively.

training data. This takes the form of an optimization problem. The Levenberg–Marquardt algorithm included in MATLAB was used due to its speed for networks of complexity similar to the ones shown here. Several metrics are used to measure the quality of the trained network. To ensure that the network is not underfit, the mean square error (MSE) is minimized during training, and to ensure that the network is not overfit, i.e., has poor generalization, the trained network is tested using data that have not been seen before. To test for overfitting without creating a separate validation data set, k -fold cross validation (KFCV) is used [20].

For the models in this paper, no more than 350 iterations were used to achieve an MSE accuracy of less than 1×10^{-4} . To improve the network's convergence rate, the range of the input data is prescaled. In particular, the Peclet number is varied on a logarithmic rather than linear scale, and the variance training data for the gated-cross injector are normalized to unity. To validate the NN models, first, the MSE and KFCV are examined; the results are summarized in Table III, with very small validation parameters throughout. As a visual confirmation, response surfaces are shown for portions of the π -space for each injector. Fig. 6 shows the results for the cross and the double-tee injector, and Fig. 7 shows the results for the gated-cross injector.

The graphs at the top represent the sparse set of points simulated in FEMLAB, while those at the bottom show the densely simulated points using the much faster NN after being trained on the FEMLAB data. The NN operates in a fraction of a second per point, more than four orders of magnitude faster than numerical simulation.

Fig. 6(a) and (b) shows the variance and effective peak concentration results, respectively, for the cross, and Fig. 6(c) and (d) shows the same results for the double-tee. When the cross and the double-tee

are operated with the same accessory fields, the cross has a smaller variance and a smaller peak concentration.

Fig. 7(a) shows the feasibility regions for the gated-cross, and Figs. 7(b) and 6(c) show the variance and peak concentration results for the gated-cross injector, respectively. The feasibility space is independent of π_{4G} , because the leakage is determined in the loading stage independently of how the band is dispensed. The most notable feature of the feasibility space is that as π_{3G} , the Peclet number, becomes smaller, the amount of feasible space decreases significantly. This is due to the fact that regardless of how much the gate is closed, when diffusion is large, the diffusive flux leaking into the separation channel is large. Fig. 7(b) shows the variance results in the π -space of the gated-cross, and Fig. 7(c) shows the effective peak concentration results. The missing surface areas in Fig. 7(b) and (c) represent infeasible regions as described in Section III-A.

D. Extracting an Analytic Equation

Since the NN is connected in a feed-forward configuration, it is straightforward to identify the explicit relation for the output node in terms of all the nodes in the previous layers. The general explicit form for the two-layer topology with a single output node is

$$y_k = g_{\text{out}} \left[\sum_{j=0}^M v_{kj} g_{\text{hid}} \left[\sum_{i=0}^N w_{ji} x_i \right] \right] \quad (10)$$

where y_k is the output of the network, $g_{\text{out}}[\xi]$ is the linear output node activation function, $g_{\text{hid}}[\xi]$ is the nonlinear hidden node activation

function from (9), and v_{kj} and w_{ji} are the weights between the hidden-output and input-hidden layers, respectively.

Once the network is trained and the weights are fixed, a simple algorithm can be written to extract the network data from the MATLAB data structures. The result is then used in VerilogA to perform analytical system simulations.

E. Injector Model Impact

As a simple example to illustrate the benefit of using the accurate NN model, a standard cross separation is performed in Fig. 8. Comparison is made between numerical simulation of the separation using FEMLAB, analytical simulation using our VerilogA simulator with the device models of the injector and separation channel, and a VerilogA simulation using a simplified injector model. The simplified injector model assumes that the resulting injection will be a Gaussian approximation to a rectangular plug with concentration C_o and standard deviation equal to half the channel's width, $w/2$. The Gaussian will have the same standard deviation as the rectangular plug and a peak height that conserves area, $C_p = C_o/\sqrt{\pi/2}$.

To compare the simulation results, the resolution of the separation is calculated. Resolution is defined as

$$R = \frac{(d_2 - d_1)}{4\sigma_{ave}} \quad (11)$$

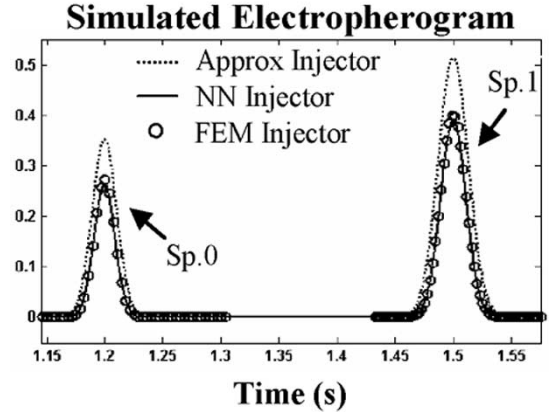
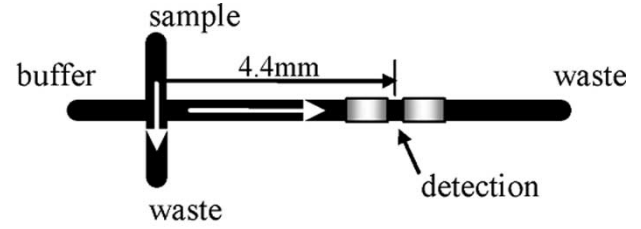
where d_2 and d_1 are the centers of the two bands in time, respectively, and σ_{ave} is the average standard deviation of the two bands in time. These quantities are meaningful when the electropherogram peaks are approximated to be Gaussian, as they are in this case. Four times the standard deviation, therefore, accounts for more than 95% of the total mass of the band. A desirable resolution is usually at least 1.5, so the ease of the separation in the example of Fig. 8 is illustrated by its very high resolution number.

The results show the simulation with the NN injector model in close agreement with the numerical simulation. The simulation using the approximate injector model shows more than 25% difference in resolution. In this system, the two bands are still easily resolved, however, in realistic systems with tens to hundreds of species, this difference becomes much more significant.

IV. CONCLUSION

This work has shown a methodology that was used to model the injector component of a microfluidic LoC, and examples were shown for the cross, the double-tee, and the gated-cross injector designs. The key contributions of this work were the use of nondimensionalized reduced variable sets for the numerical simulation, and the use of simplified Gaussian interface parameters to describe the injection output to the downstream separation channel models. The impact of the NN-based injector model was seen to be significant compared to a simple approximate injector model and could become a significant design factor in realistic systems separating tens to hundreds of species. These injector models combined with other component models allow for a functional block decomposition of the system for efficient simulation. The speed and accuracy of these analytic block models present a far more feasible method of computer-aided design (CAD) than using numerical solutions of partial differential equations (PDEs) for the whole complex system.

This work lays the foundation for future enhancements that include the geometric π -space variables for a more complete synthesis formulation. Due to its ability to model arbitrary functional forms with an



Simulation Properties

	Sp. 0	Sp. 1
π_{1c}	500	469
π_{2c}	500	469
π_{3c}	.5	.5
π_{4c}	2	2
C_o	0.7	1

Resolution Results

	Res
Approx	538
NN	746
FEM	734

Fig. 8. Simulations of the cross separation system using component separation system model and 1) an approximate injector as described in the text, 2) the NN injector, 3) full finite-element method (FEM) simulation of injector and separation channel.

arbitrary number of inputs and outputs, with further investigation, the methodology described here can be used to model other microfluidic components for use in composable system simulation.

REFERENCES

- [1] D. R. Reyes, D. Iossifidis, P.-A. Auroux, and A. Manz, "Micro total analysis systems—1: Introduction, theory, and technology," *Anal. Chem.*, vol. 74, no. 12, pp. 2623–2636, Jun. 2002.
- [2] P.-A. Auroux, D. Iossifidis, D. R. Reyes, and A. Manz, "Micro total analysis systems—2: Analytical standard operations and applications," *Anal. Chem.*, vol. 74, no. 12, pp. 2637–2652, Jun. 2002.
- [3] Y. Wang, Q. Lin, and T. Mukherjee, "Applications of behavioral modeling and simulation on lab-on-a-chip: Micro-mixer and separation system," in *Behavioral Modeling and Simulation Conf.*, San Jose, CA, 2004, pp. 1–6.
- [4] R. Magargle, J. F. Hoburg, and T. Mukherjee, "An injector component model for complete microfluidic electrokinetic separation systems," in *Proc. Nanotechnology Conf. and Trade Show (NanoTech)*, Boston, MA, 2004, pp. 77–80.
- [5] Y. Wang, Q. Lin, and T. Mukherjee, "System-oriented dispersion models of general-shaped electrophoresis microchannels," *Lab Chip*, vol. 4, no. 5, pp. 453–463, Oct. 2004.
- [6] Y. Wang, R. Magargle, Q. Lin, J. F. Hoburg, and T. Mukherjee, "System-oriented modeling and simulation of biofluidic lab-on-a-chip," in *Proc. Transducers*, Seoul, Korea, 2005, vol. 2, pp. 1280–1283.

- [7] K. Seiler, Z. H. Fan, K. Fluri, and D. J. Harrison, "Electroosmotic pumping and valveless control of fluid flow within a manifold of capillaries on a glass chip," *Anal. Chem.*, vol. 66, no. 20, pp. 3485–3491, Oct. 1994.
- [8] S. C. Jacobson, R. Hergenröder, L. B. Koutny, and R. J. Warmack, "Effects of injection schemes and column geometry on the performance of microchip electrophoresis devices," *Anal. Chem.*, vol. 66, no. 7, pp. 1107–1113, Apr. 1994.
- [9] S. V. Ermakov, S. C. Jacobson, and J. M. Ramsey, "Computer simulations of electrokinetic transport in microfabricated channel structures," *Anal. Chem.*, vol. 70, no. 21, pp. 4494–4504, 1998.
- [10] L. L. Shultz-Lockyear, C. L. Colyer, Z. H. Fang, K. I. Roy, and D. J. Harrison, "Effects of injector geometry and sample matrix on injection and sample loading in integrated capillary electrophoresis devices," *Electrophoresis*, vol. 20, no. 3, pp. 529–538, Mar. 1999.
- [11] M. Deshpande, K. B. Greiner, J. West, and J. R. Gilbert, "Novel designs for electrokinetic injection in μ TAS," in *Micro Total Analysis Systems*, Enschede, The Netherlands, 2000, pp. 339–342.
- [12] S. V. Ermakov, S. C. Jacobson, and J. M. Ramsey, "Computer simulations of electrokinetic injection techniques in microfluidic devices," *Anal. Chem.*, vol. 72, no. 15, pp. 3512–3517, Jun. 2000.
- [13] H. Liu, A. Singhee, R. A. Rutenbar, and L. R. Carley, "Remembrance of circuits past: Macromodeling by data mining in large analog design spaces," in *Design Automation Conf.*, New Orleans, LA, 2002, pp. 437–442.
- [14] O. Mikulchenko, A. Rasmussen, and K. Mayaram, "A neural network based macromodel for microflow sensors," in *Proc. Modeling and Simulation Microsystems (MSM)*, San Diego, CA, 2000, pp. 540–543.
- [15] P. Bratley and B. L. Fox, "Implementing Sobol's quasirandom sequence generator," *ACM Trans. Math. Softw.*, vol. 14, no. 1, pp. 88–100, Mar. 1998.
- [16] S. Rudolph, "On topology, size and generalization of non-linear feed-forward neural networks," *Neurocomputing*, vol. 16, no. 1, pp. 1–22, Jul. 1997.
- [17] Y. Lee, Y. Park, F. Niu, B. Bachman, and D. Filipovic, "Computer aided design and optimization of integrated circuits with RF MEMS devices by an ANN based macro-modeling approach," in *Proc. Modeling and Simulation Microsystems (MSM)*, Anaheim, CA, 2005, vol. 3, pp. 565–568.
- [18] N. A. Patankar and H. H. Hu, "Numerical simulation of electroosmotic flow," *Anal. Chem.*, vol. 70, no. 9, pp. 1870–1881, May 1998.
- [19] The Mathworks. [Online]. Available: <http://www.mathworks.com>
- [20] C. M. Bishop, *Neural Networks for Pattern Recognition*. New York: Oxford Univ. Press, 1995.
- [21] Comsol Multiphysics. [Online]. Available: <http://www.comsol.com>
- [22] Y. Wang, Q. Lin, and T. Mukherjee, "A model for joule heating-induced dispersion in microchip electrophoresis," *Lab Chip*, vol. 4, no. 6, pp. 625–631, Dec. 2004.
- [23] N. H. Chiem and D. J. Harrison, "Microchip systems for immunoassay: An integrated immunoreactor with electrophoretic separation for serum theophylline determination," *Clin. Chem.*, vol. 44, no. 3, pp. 591–598, Mar. 1998.
- [24] A. R. Barron, "Universal approximation bounds for superpositions of a sigmoidal function," *IEEE Trans. Inf. Theory*, vol. 39, no. 3, pp. 930–945, May 1993.